Machine Learning Prediction for Classification of Outcomes in Local Minimisation

Ritankar Das[1] and David J. Wales[1, a)]

University Chemical Laboratories, Lensfield Road, Cambridge CB2 1EW, United Kingdom

Machine learning schemes are employed to predict which local minimum will result from local energy minimisation of random starting configurations for a triatomic cluster. The input data consists of structural information at one or more of the configurations in sequences that begin at random geometries and converge to one of four distinct local minima. Two of the three interatomic distances are used, with systematic comparisons of the predictions obtained using sequences of configurations starting from different positions in the optimisation pathways. The objective is to predict which of the four possible outcomes will result within a given tolerance, at the lowest possible computational cost. The ability to make reliable predictions, in terms of the energy or other properties of interest, could save significant computational resources in sampling procedures that involve systematic geometry optimisation. Results are compared for two energy minimisation schemes, and for neural network and quadratic fitting functions. We find that prediction quality is mostly determined by the configuration that lies closest to convergence for the neural networks, but there can be some benefit in using sequences of configurations for the quadratic function. Accurate fits can be precluded if the regularisation parameter is too large.

a)dw34@cam.ac.uk

I. INTRODUCTION

Several recent contributions have addressed the application of methods developed to explore molecular potential energy landscapes[1–3] to the solution space defined by the parameters of a neural network.[4–7] Fitting these parameters to a set of training data, where each entry has an associated outcome, produces a predictive classification tool belonging to the field of machine learning.[8,9] The potential energy landscape employed in molecular science is replaced by the machine learning (ML) landscape of the objective function that is minimised in the fitting procedure. This objective function is generally defined in terms of the difference between the predicted and true outcomes, corresponding to a cost (or loss) function. Non-convex cost functions can generally exhibit a range of local minima,[10–13] and the connections between them via transition states (saddle points of Hessian index one) define the connectivity of a transition network.[3,14,15]

We have described the application of energy landscape methodology based on previous work for molecular structure, dynamics, and thermodynamics in previous reports, where more details can be found.[6,7] In particular, we have examined how features of the corresponding landscape defined by a machine learning procedure are connected to the appearance of thermodynamic analogues, such as the heat capacity.

The present work builds upon our initial study of geometry optimisation for a triatomic cluster[6,7] to focus on the effect of memory. Machine learning is used to predict which local minimum will be found from the set of possible solutions, using one or more configurations from an energy minimisation sequence. This is a classification problem in terms of a known solution set, as opposed to ab initio structure prediction. The relevant applications involve repeated local minimisation to associate points in phase space with particular local minima, or with local minima corresponding to energies or other properties of interest that lie in specific ranges (bins). In particular, the basin-sampling approach to global thermodynamics provides a powerful way to overcome broken ergodicity,[16] exploiting regular local minimisations to generate statistics for a two-dimensional array of instantaneous and quench energy bins. Accurate thermodynamics have been obtained for benchmark atomic clusters, where the equilibrium structure changes at temperatures for which the intervening barrier is very large compared to the available thermal energy. Systematic minimisation has also been used to calculate the absolute volume of basins of attraction for local minima corresponding to three-dimensional jammed packings, and hence access measures of configurational entropy in granular packings.[17] The ability to terminate geometry

optimisation earlier to save computational resources,[18] while retaining sufficient confidence in the classification of the resulting minimum, has the potential to accelerate sampling significantly.

Before considering applications to larger systems that pose significant challenges for sampling global equilibrium thermodynamics, we wish to explore and provide benchmarks for some simpler molecules. The present results further this aim in several ways. To make the classification more challenging we employ two of the interatomic distances, rather than all three. To investigate the effect of memory, or correlations, in the input data we analyse the effect of including successive configurations. Using such sequences might help to highlight the key geometrical parameters that determine the classification, and average out noise in the data. We also consider results for two energy minimisation algorithms and two fitting functions for the machine learning component.

The machine learning classifications require the possible outcomes to be known in terms of the available local minima. For the simple test system considered in these benchmark calculations there are four possible results, and we employ fitting parameters obtained in training to make predictions for configurations that appear anywhere in minimisation sequences for a separate testing database. Since these minimisations were also followed to convergence, we can quantify the accuracy of the predictions by calculating area under receiver operating characteristic plots, as described in §IV. In practical applications the geometry optimisation would be terminated before convergence, and a prediction would be made for the classification. For complex systems it may be useful to attempt classification to bins where some property of interest lies in a specified range. To analyse thermodynamics this property would probably be the potential energy, but it may also be useful to classify predictions in terms of structural order parameters that distinguish different morphologies, such as close-packed or icosahedral motifs for atomic clusters.[16,19,20]

Our results indicate that for neural network fits, the reliability of the predictions for the outcome of local minimisation depends mostly upon how close the latest known configuration is to convergence, and that including memory of previous configurations in the minimisation sequence does not have much effect. This conclusion holds for both the geometry optimisation schemes considered. However, a potentially significant systematic improvement in prediction quality is observed when the neural network fit is replaced by a quadratic function of all the inputs (§V). We also note the importance of choosing a regularisation parameter that is small enough not to exclude regions of parameter space that support the most accurate fits.

II. GEOMETRY OPTIMISATION FOR A MODEL TRIATOMIC CLUSTER

The cluster considered here has previously served as a benchmark to visualise and compare the performance of different geometry optimisation techniques.[21–23] The interatomic potential is a sum of pairwise Lennard-Jones terms[24] and a three-body Axilrod–Teller contribution,[25] which corresponds to an instantaneous induced dipole-induced dipole interaction. Here it simply represents a convenient way to tune the number of local minima on the molecular potential energy surface, defined by

$$V = 4\varepsilon \sum_{i<j} \left[\left(\frac{\sigma}{r_{ij}} \right)^{12} - \left(\frac{\sigma}{r_{ij}} \right)^{6} \right] + Z \sum_{i<j<k} \left[\frac{1 + 3\cos\theta_1 \cos\theta_2 \cos\theta_3}{(r_{ij} r_{ik} r_{jk})^3} \right], \qquad (1)$$

where θ_1, θ_2 and θ_3 are the internal angles of the triangle formed by atoms i, j, k. r_{ij} is the distance between atoms i and j, and the parameter Z scales the magnitude of the three-body term. All the present results correspond to $Z = 2$, for which there are three linear local minima (potential energy $-2.219\,\varepsilon$), distinguished according to which atom is in the central position, and one minimum for an equilateral triangle (potential energy $-2.185\,\varepsilon$). The interparticle distances in the triangle are all 1.16875 (in units of σ), and in the linear minima the two nearest-neighbour distances are 1.10876.

To distinguish between the energy landscape defined by stationary points of the cluster, and the landscape defined by stationary points of the ML objective function, we will refer to the *molecular energy landscape* and the *ML landscape*, respectively.

We first considered a database of 10,000 local minimisation sequences starting from random coordinates in a cube of side length $2\sqrt{3}$ reduced units, as described in our previous report.[6] 5,000 sequences (chosen at random) were used to train the networks, and the other 5,000 were used for testing. To create this database a customised LBFGS routine (a limited memory version of the quasi-Newton Broyden,[26] Fletcher,[27] Goldfarb,[28] Shanno,[29] BFGS procedure) was used, with a convergence condition on the root mean square gradient of 10^{-6} reduced units. The outcome of each geometry optimisation is one of the four local minima, which are numbered from 0 to 3: 0 refers to the triangle, and 1, 2 and 3 refer to the linear minima with atoms 1, 2 and 3 in the centre position. Our aim is to predict this result based on information from one or more of the intermediate configurations in the series of steps leading from the initial high energy structure to a minimum. Specifically, in this study, we investigate whether more accurate predictions can be obtained by including data from sequences of configurations from each minimisation, and how the results depend upon where the configurations are taken from, in terms of the number of steps from

convergence.

From the structural and energetic properties of each configuration we chose the two distances r_{12} and r_{13} for training and testing the neural networks. This choice will make it harder to distinguish outcomes 0 and 1, since the corresponding distances only differ by about 5% in the two minima. The number of geometry optimisation steps in the minimisation sequences in the database varies from 11 to 91. To define a convenient input format we extracted r_{12} and r_{13} at each step. We stored these inputs for the configuration corresponding to the converged minimum at position 1, for the configuration before convergence at position 2, etc., up to the r_{12} and r_{13} values for the initial random starting configuration, which appeared between position 11 and position 91, depending on the minimisation sequence. We duplicated the entries for the random starting configuration where necessary, so that each sequence contained 91 entries for r_{12} and r_{13}. This padding corresponds to a worst case scenario for sequences shorter than the number of steps to convergence under consideration.

Training and testing results were obtained for input data consisting of r_{12} and r_{13} values at one, two and three successive configurations, varying the first entry systematically from index one (the converged minimum) to entry 80. For example, in the runs based on three steps, the results beginning from step 80 involve configurations 80, 81 and 82. As described above, for minimisation sequences with fewer than 80 steps these three entries each correspond to r_{12} and r_{13} at the random starting configuration. Hence the influence of a short minimisation sequences will not change when the first step considered is greater than or equal to the total number of steps in that series. There are certainly other ways to deal with the different sequence lengths, but we do not believe our principal conclusions would be affected.

To provide further insight we have compared results obtained using the Bulirsch-Stoer algorithm for minimisation, instead of LBFGS, and we have tested a quadratic fitting function for comparison with the neural network (described in §III). The Bulirsch-Stoer algorithm provides a numerical solution of the steepest-descent equations using rational function Richardson extrapolation and the modified midpoint method.[30–33] Optimisations were initiated from the same set of 10,000 starting points as for the LBFGS runs, and 9,000 successful minimisation sequences that converged to a root mean square gradient of 10^{-5} reduced units in 1,500 steps or less were used to create a new database. The number of steps required is typically at least an order of magnitude more than for LBFGS, as expected for a steepest-descent approach.[23] As for the LBFGS results, the configurations were indexed with the final converged configuration in position 1 running through

to the starting configuration and then duplicating that structure, if required, to produce 1,500 entries for each sequence. In this case we tested memory effects by considering blocks of one, ten, and twenty successive configurations, starting from all possible positions in each minimisation sequence. The larger block sizes are required for comparison with the LBFGS results, because the step length in the Bulirsch-Stoer sequences is typically an order of magnitude smaller. The training and testing sequences consisted of non-overlapping sets of 4,500 sequences chosen at random from the database of 9,000 optimisations. In this case we considered one value for the regularisation parameter λ of 10^{-5}. The training results for the three different block sizes positioned at all possible distances from convergence in the sequences of 1,500 configurations required nearly 18,000 basin-hopping global optimisation runs. The same parameters were used for these basin-hopping surveys of the ML landscape as for the LBFGS training data.

III. NEURAL NETWORK AND QUADRATIC FITS

Our initial investigations of ML landscapes have employed artificial neural networks, since the corresponding objective function has analytic derivatives, and has been used quite widely in molecular science. The networks consisted of $N_{\rm in} = 2$, 4, or 6 inputs for data constructed using r_{12} and r_{13} at one, two and three successive steps in each geometry optimisation sequence for the LBFGS minimisation sequences of configurations. Each output layer contained $N_{\rm out} = 4$ nodes (or classes), for the four local minima. Results were compared for a single hidden layer consisting of three, four, five, and six nodes. A single hidden layer is known to be sufficient for a wide variety of applications,[8] and the overall structure follows the architecture used by Brown, Gibbs and Clary in their application to fitting intermolecular potentials.[34]

The inputs can be written as $\mathbf{Z} = \{\mathbf{z}^1, \ldots, \mathbf{z}^{N_{\rm data}}\}$, with each component having dimension $N_{\rm in}$, so that $\mathbf{z}^\alpha = \{z_1^\alpha, \ldots, z_{N_{\rm in}}^\alpha\}$. The output at node i is

$$y_i^{\rm NN} = \phi_2 \left[\sum_{j=1}^{N_{\rm hidden}} \left(w_{ij}^{(1)} + w_o^b \right) \phi_1 \left[\sum_{k=1}^{N_{\rm in}} \left(w_{jk}^{(2)} + w_h^b \right) z_k \right] \right], \tag{2}$$

with activation functions $\phi_2(\xi) = \tanh(\xi)$ and $\phi_1(\xi) = \xi$, bias values w_h^b and w_o^b, and link weights $w_{ij}^{(1)}$ between hidden node j and output i, and $w_{jk}^{(2)}$ between input k and hidden node j.

The four output values, $\mathbf{y}^{\rm NN} = \{y_0^{\rm NN}, y_1^{\rm NN}, y_2^{\rm NN}, y_3^{\rm NN}\}$, were converted to softmax probabilities,

which reduce the influence of outliers:

$$p_c^{\mathrm{NN}}(\mathbf{W};\mathbf{Z}) = e^{y_c^{\mathrm{NN}}} / \sum_{a=0}^{3} e^{y_a^{\mathrm{NN}}}. \tag{3}$$

Training was performed for $N_{\mathrm{data}} = 5,000$ of the LBFGS minimisation sequences. The resulting objective function included an L2 regularisation term, intended to prevent overfitting, proportional to a parameter λ, to give

$$E^{\mathrm{NN}}(\mathbf{W};\mathbf{Z}) = -\frac{1}{N_{\mathrm{data}}} \sum_{d=1}^{N_{\mathrm{data}}} \ln p_{c(d)}^{\mathrm{NN}}(\mathbf{W};\mathbf{Z}) + \lambda \left(\sum_{j=1}^{N_{\mathrm{hidden}}} \sum_{i=0}^{3} \left(w_{ij}^{(1)}\right)^2 + \sum_{j=1}^{N_{\mathrm{hidden}}} \sum_{k=1}^{N_{\mathrm{in}}} \left(w_{jk}^{(2)}\right)^2 \right), \tag{4}$$

Here $c(d) = 0$, 1, 2 or 3 is the class label for data point d specified by the minimum obtained in training set optimisation d, and the components of $\mathbf{W}$ are the variables $w_{ij}^{(1)}$, $w_{jk}^{(2)}$, w_o^b and w_h^b. Guided by the previous study,[6] results were compared for $\lambda = 10^{-3}$, 10^{-4} and 10^{-5}. For testing, the parameters $\mathbf{W}$ are fixed at the fitted values obtained by minimising $E^{\mathrm{NN}}(\mathbf{W};\mathbf{Z})$ for specific training data, $\mathbf{Z}$. The output bias w_o^b was not actually used here, since it has no effect for the chosen activation functions.

This ML objective function and derivatives have been implemented in the `GMIN` global optimisation program[35] and the `OPTIM` code for analysing stationary points and pathways.[36] Minimisation of the ML objective function $E^{\mathrm{NN}}(\mathbf{W};\mathbf{Z})$, employed the same customised LBFGS routine that we used to create the initial database of geometry optimisation sequences.

To provide a comparison with the neural network fits an alternative quadratic function was considered for each output:

$$y_i^{\mathrm{Q}} = w^{(0)}(i) + \sum_{k=1}^{N_{\mathrm{in}}} w_k^{(1)}(i) z_k + \sum_{k=1,j\geq k}^{N_{\mathrm{in}}} w_{kj}^{(2)}(i) z_k z_j. \tag{5}$$

This formulation involves a fitting function with $N_{\mathrm{out}}(1+N_{\mathrm{in}}(N_{\mathrm{in}}+3)/2)$ variables. The calculated outputs were then used to produce softmax probabilities p_i^{Q} analogous to equation (3), with an objective function

$$E^{\mathrm{Q}}(\mathbf{W};\mathbf{Z}) = -\frac{1}{N_{\mathrm{data}}} \sum_{d=1}^{N_{\mathrm{data}}} \ln p_{c(d)}^{\mathrm{Q}}(\mathbf{W};\mathbf{Z}) + \lambda \sum_{i=0}^{3} \left(\left[w^{(0)}(i)\right]^2 + \sum_{k=1}^{N_{\mathrm{in}}} \left[w_k^{(1)}(i)\right]^2 + \sum_{k=1,j\geq k}^{N_{\mathrm{in}}} \left[w_{kj}^{(2)}(i)\right]^2 \right), \tag{6}$$

analogous to equation (4). Analytic first and second derivatives have again been implemented in `GMIN` and `OPTIM`. The quadratic fit was used to provide a comparison for the database of LBFGS optimisation sequences, as described in §V.

IV. EXPLORING THE ML LANDSCAPE AND QUANTIFYING PREDICTIONS

The ML landscape was explored for each set of training data using basin-hopping (BH) global optimisation,[37–39] as described in previous work.[6,7] Locating the global minimum is straightforward for the systems considered here, and we chose basin-hopping parameters to provide a wider sample of low-lying minima for $E(\mathbf{W};\mathbf{Z})$, to check the quality of predictions obtained using alternative fits. The BH runs each consisted of at least 1,000 BH steps, and some were repeated for 10,000 steps to check that the minimum producing the best predictions for each testing set had been included, which was always found to be the case. Checks were also conducted for different BH step size and temperature parameters, none of which changed the results reported below.

To compare the predictions we studied the area under curve (AUC) for receiver operating characteristic (ROC) plots. The ROC curves are plots of the true positive rate, $T_{\rm pr}$, against the false positive rate, $F_{\rm pr}$, as a function of the threshold probability, P, for predicting convergence to the equilateral triangle, rather than one of the linear minima. Hence

$$\begin{aligned} T_{\rm pr}(\mathbf{W};P) &= \sum_{d=1}^{N_{\rm data}} \delta_{c(d),0}\Theta(p_0(\mathbf{W})-P)\Big/\sum_{d=1}^{N_{\rm data}}\delta_{c(d),0}, \\ F_{\rm pr}(\mathbf{W};P) &= \sum_{d=1}^{N_{\rm data}} (1-\delta_{c(d),0})\Theta(p_0(\mathbf{W})-P)\Big/\sum_{d=1}^{N_{\rm data}}(1-\delta_{c(d),0}), \end{aligned} \tag{7}$$

where Θ is the Heaviside step function and δ is the Kronecker delta. The AUC values were obtained by numerical integration of

$$\mathrm{AUC}(\mathbf{W}) = \int_0^1 T_{\rm pr}(\mathbf{W};P)dF_{\rm pr}(\mathbf{W};P). \tag{8}$$

The AUC value is interpreted as the probability that for two optimisation series chosen at random from the sets that converge to the triangle and to a linear geometry, the prediction will discriminate between them correctly. The predictions are roughly classified as 'fair' for AUC values between 0.7 and 0.8, 'good' in the range 0.8 to 0.9, and 'excellent' in the range 0.9 to 1.

V. RESULTS

The key results for the neural network formulation are illustrated in Figures 1 and 2. The first of these Figures shows the AUC values corresponding to the global minimum neural network parameters obtained with the training set of 5,000 minimisation sequences when applied to the

testing data set containing the remaining 5,000 sequences. For regularisation parameters $\lambda = 10^{-4}$, 10^{-5} and 10^{-6}, results are compared for 3, 4, 5, and 6 hidden nodes using r_{12} and r_{13} values at one, two and three consecutive points from each minimisation sequence. The first configuration in each sequence was systematically varied from configuration 1 (the converged minimum) to configuration 80. For example, the AUC values at position 10 on the horizontal axis correspond to input data (r_{12} and r_{13}) for the 10th, 10th and 11th, and 10th, 11th and 12th configurations in each sequence. Results are shown up to configuration 60, beyond which there is little variation.

The behaviour for the largest regularisation parameter considered, $\lambda = 10^{-4}$, is qualitatively different for data corresponding to configurations near convergence. Here we would hope to make accurate predictions, and this is indeed the case for the smaller values of λ, and for some of the results with $\lambda = 10^{-4}$. However, with only three hidden nodes, the predictions are actually worse in this limit. This behaviour appears to be somewhat mitigated when more hidden nodes are used, or when memory of previous configurations is included.

For the two smaller values of λ the results are more stable. Here, the most important observation is that there is little difference in prediction quality when sequences of configurations are included in the training data. For the neural networks, it is the configuration closest to convergence that mainly determines the accuracy of predictions for the testing data. Hence the additional computational expense of including data corresponding to additional memory of previous configurations does not appear worthwhile for these fitting functions.

To check that the neural network parameters of the global minimum obtained with the training set are an optimal choice for the testing data, the AUC values were calculated using all the applicable local minima from the basin-hopping surveys with the training data. The highest AUC value obtained for the testing database and training fits corresponding to the same number of hidden nodes with all three values of λ is compared with the results for the global minimum in Figure 2. Here the training data consisting of a single configuration was used, and solutions obtained for configurations taken from all positions in the minimisation sequences from 1 to 80 are admitted. Since these AUC values include the result for the global minimum, the resulting maximum value AUC curves must always lie on or above the curve corresponding to that solution. Figure 2 shows that the global minimum does indeed provide predictions that are close to optimal, especially in the most important range corresponding to higher accuracy. This result demonstrates that basin-hopping global optimisation is an effective way to locate a solution that is unlikely to be improved significantly for the testing set by any other local minimum, in agreement with observations for

very different applications.[40]

The limiting AUC values for configurations progressively further from convergence are around 0.7, which is quite a high baseline. This result indicates that there is some minimal amount of information about the resulting classification in the starting configurations. It would be interesting to check whether this baseline shifts when the container size for the random initial coordinate distribution is changed, and we plan to investigate this possibility in future work.

The results for neural network fits with the Bulirsch-Stoer steepest-descent algorithm are shown in Figure 3, where blocks of one, ten, and twenty successive configurations have been considered as input data starting from every possible position in each sequences, as indexed along the horizontal axis. The left panels of this Figure show the AUC values obtained with the testing data for the neural network weights fixed at the values for the global minimum obtained in the training phase. The panels on the right show the maximum AUC values obtained for any local minimum of the ML training landscape. The latter plots are less noisy, but show the same general trends. In particular, the results for fits based upon inputs that include one, two, and three successive sets of r_{12} and r_{13} values are very similar. However, as the inputs shift further to the right on the horizontal axis, away from convergence, it is clear that there exist local minima for the training function that provide better AUC values than the training global minimum. This result suggests that there may be scope for improved predictions by combining different solutions for the training ML landscape, which also merits further investigation.

Our final comparison employs the quadratic fitting function for the LBFGS minimisation sequences. Here we considered input data consisting of between one and ten successive r_{12} and r_{13} values, again taken from all possible starting points in the testing data. The AUC values obtained using the parameters corresponding to the global minimum for the training data are shown in Figure 4. The number of variables defining the ML landscape increases from 24 to 924 for one and ten r_{12} and r_{13} input values, respectively. Here we see a systematic improvement in the AUC values for predictions based on more inputs in the intermediate region. Hence, in this formulation it may be beneficial to include some memory effects in training the ML fitting function, which suggests that it may be possible to exploit correlations between successive configurations.

VI. CONCLUSIONS

The ability of machine learning fits to predict the outcome of energy minimisation in terms of the possible local minima has been investigated for a simple triatomic molecule. The effect of varying the number of successive configurations used as input data is examined, using two of the three interparticle distances at successive steps in the geometry optimisation sequence. This choice makes it more difficult to discriminate one of the three linear minima from the equilateral triangle, because the two distances in question are 1.1087 and 1.1687 in the two structures, respectively. We have compared results for geometry optimisation sequences obtained with an LBFGS minimiser and the Bulirsch-Stoer steepest-descent approach, and for neural network and quadratic fitting functions.

The quality of the classification predictions mainly depends on the configuration closest to convergence for the neural network fits. Using additional data corresponding to previous configurations in the minimisation sequence does not have much effect. Hence it does not seem worthwhile to increase the dimensionality of the fitting procedure and the network to include additional configurations beyond the most recent one in this framework. To investigate whether this result reflects the use of previous gradients and steps in constructing the approximate inverse Hessian update for the LBFGS approach, we compared results obtained using Bulirsch-Stoer minimisation. As expected,[23] many more iterations are required to achieve convergence using steepest-descent, and including memory in terms of sequential configurations as input data again has little effect for the neural network fits.

Since the LBFGS procedure used here has proved to be particularly powerful in previous benchmarks[41] this formulation corresponds to the framework that is most likely to be employed in applications. For the neural network fits, accurate predictions are obtained for configurations within about 20 steps of convergence, and beyond about 30 steps there is little further variation. Furthermore, the predictions obtained using parameters corresponding to the global minimum for the training data are generally close to optimal for the LBFGS minimisation sequences, compared to alternative low-lying minima sampled during basin-hopping global optimisation surveys. However, for the Bulirsch-Stoer steepest-descent approach it may be beneficial to combine predictions from local minima of the training landscape. For both minimisation algorithms the accuracy of predictions, and the degradation of AUC values as the testing data shifts further from convergence, does not change significantly when more hidden nodes are used, so long as λ is not too large. How-

ever, a systematic improvement does appear for the LBFGS sequences when longer sequences of configurations are used if a quadratic fitting function is used in place of the neural nets. This result may reflect the quadratic rather than linear increase in the number of fitting parameters with the number of inputs, and further investigation appears warranted.

We finally note that if the regularisation parameter λ, which is intended to reduce overfitting, is too large, accurate fits can be excluded. Using more hidden nodes or more input data, in the form of successive configurations, can partly counter this effect, probably because the number of variables involved in optimising the network increases. Systematic approaches to the choice of regularisation parameters, such as Bayesian analysis,[42] might provide additional insight. There will be more local minima when λ is smaller,[6] so locating the global minimum reliably will require more basin-hopping steps. However, alternative low-lying minima for the training data often provide predictions of comparable accuracy. Hence it may be best to choose a smaller value, to ensure that accurate solutions are not excluded by regularisation. Future work will investigate to what extent all these observations carry over to more complex systems.

REFERENCES

[1]D. J. Wales, Energy Landscapes, Cambridge University Press, Cambridge, 2003.

[2]D. J. Wales, Phil. Trans. Roy. Soc. A 370 (2012) 2877–2899.

[3]D. J. Wales, Curr. Op. Struct. Biol. 20 (2010) 3–10.

[4]M. Pavlovskaia, K. Tu, S.-C. Zhu, arXiv:1410.0576 [stat.ML].

[5]A. J. Ballard, J. Stevenson, D. J. Wales (2016).

[6]A. J. Ballard, J. D. Stevenson, R. Das, D. J. Wales, J. Chem. Phys. 144 (2016) 124119.

[7]R. Das, D. J. Wales, Phys. Rev. E 93 (2016) 063310.

[8]T. Hastie, R. Tibshirani, J. Friedman, The Elements of Statistical Learning, Springer, New York, 2009.

[9]C. M. Bishop, Pattern Recognition and Machine Learning, Springer, New York, 2006.

[10]R. Collobert, F. Sinz, J. Weston, L. Bottou, Trading Convexity for Scalability, ICML '06, ACM, New York, NY, USA, 2006.

[11]L. Zhao, M. A. Mammadov, J. Yearwood, From Convex to Nonconvex: A Loss Function Analysis for Binary Classification, 2010.

[12]A. Choromanska, M. Henaff, M. Mathieu, G. B. Arous, Y. LeCun, CoRR abs/1412.0233 (2014).

URL: http://arxiv.org/abs/1412.0233.

[13]Y. Dauphin, R. Pascanu, Ç. Gülçehre, K. Cho, S. Ganguli, Y. Bengio, CoRR abs/1406.2572 (2014). URL: http://arxiv.org/abs/1406.2572.

[14]F. Noé, S. Fischer, Curr. Op. Struct. Biol. 18 (2008) 154–162.

[15]D. Prada-Gracia, J. Gómez-Gardenes, P. Echenique, F. Fernando, PLoS Comput. Biol. 5 (2009) e1000415.

[16]D. J. Wales, Chem. Phys. Lett. 584 (2013) 1 – 9.

[17]S. Martiniani, K. J. Schrenk, J. D. Stevenson, D. J. Wales, D. Frenkel, Phys. Rev. E 93 (2016) 012906.

[18]K. Swersky, J. Snoek, R. P. Adams, arXiv:1406.3896 [stat.ML] (2014).

[19]P. J. Steinhardt, D. R. Nelson, M. Ronchetti, Phys. Rev. B 28 (1983) 784.

[20]J. S. van Duijneveldt, D. Frenkel, J. Chem. Phys. 96 (1992) 4655.

[21]D. J. Wales, J. Chem. Soc. Faraday Trans. 88 (1992) 653–657.

[22]D. J. Wales, J. Chem. Soc. Faraday Trans. 89 (1993) 1305–1313.

[23]D. Asenjo, J. D. Stevenson, D. J. Wales, D. Frenkel, J. Phys. Chem. B 117 (2013) 12717–12723.

[24]J. E. Jones, A. E. Ingham, Proc. R. Soc. A 107 (1925) 636–653.

[25]P. M. Axilrod, E. Teller, J. Chem. Phys. 11 (1943) 299.

[26]C. G. Broyden, J. Inst. Math. Appl. 6 (1970) 76–90.

[27]R. Fletcher, Comput. J. 13 (1970) 317–322.

[28]D. Goldfarb, Math. Comput. 24 (1970) 23–26.

[29]D. F. Shanno, Math. Comput. 24 (1970) 647–656.

[30]R. Bulirsch, S. Stoer, Numerische Mathematik 8 (1966) 1–13.

[31]R. Bulirsch, S. Stoer, Numerische Mathematik 8 (1966) 93–104.

[32]F. S. Acton, Numerical Methods that Work, Washington: Mathematical Association of America, 1990, p. 204.

[33]W. H. Press, S. A. Teukolsky, W. T. Vetterling, B. P. Flannery, Numerical Recipes in FORTRAN, 2nd edition ed., Cambridge University Press, 1992.

[34]D. F. R. Brown, M. N. Gibbs, D. C. Clary, J. Chem. Phys. 105 (1996) 7597.

[35]D. J. Wales, Gmin: A program for basin-hopping global optimisation, basin-sampling, and parallel tempering, http://www-wales.ch.cam.ac.uk/software.html.

[36]D. J. Wales, Optim: A program for geometry optimisation and pathway calculations, http://www-wales.ch.cam.ac.uk/software.html.

[37]Z. Li, H. A. Scheraga, Proc. Natl. Acad. Sci. USA 84 (1987) 6611–6615.

[38]Z. Li, H. A. Scheraga, J. Mol. Struct. 179 (1988) 333.

[39]D. J. Wales, J. P. K. Doye, J. Phys. Chem. A 101 (1997) 5111–5116.

[40]A. Choromanska, M. Henaff, M. Mathieu, G. Ben Arous, Y. LeCun, CoRR abs/1412.0233 (2014). URL: http://arxiv.org/abs/1412.0233.

[41]S. T. Chill, J. Stevenson, V. Ruehle, C. Shang, P. Xiao, J. D. Farrell, D. J. Wales, G. Henkelman, J. Chem. Theor. Comput. 10 (2014) 5476–5482.

[42]D. MacKay, Neural Computation 4 (1992) 415447.

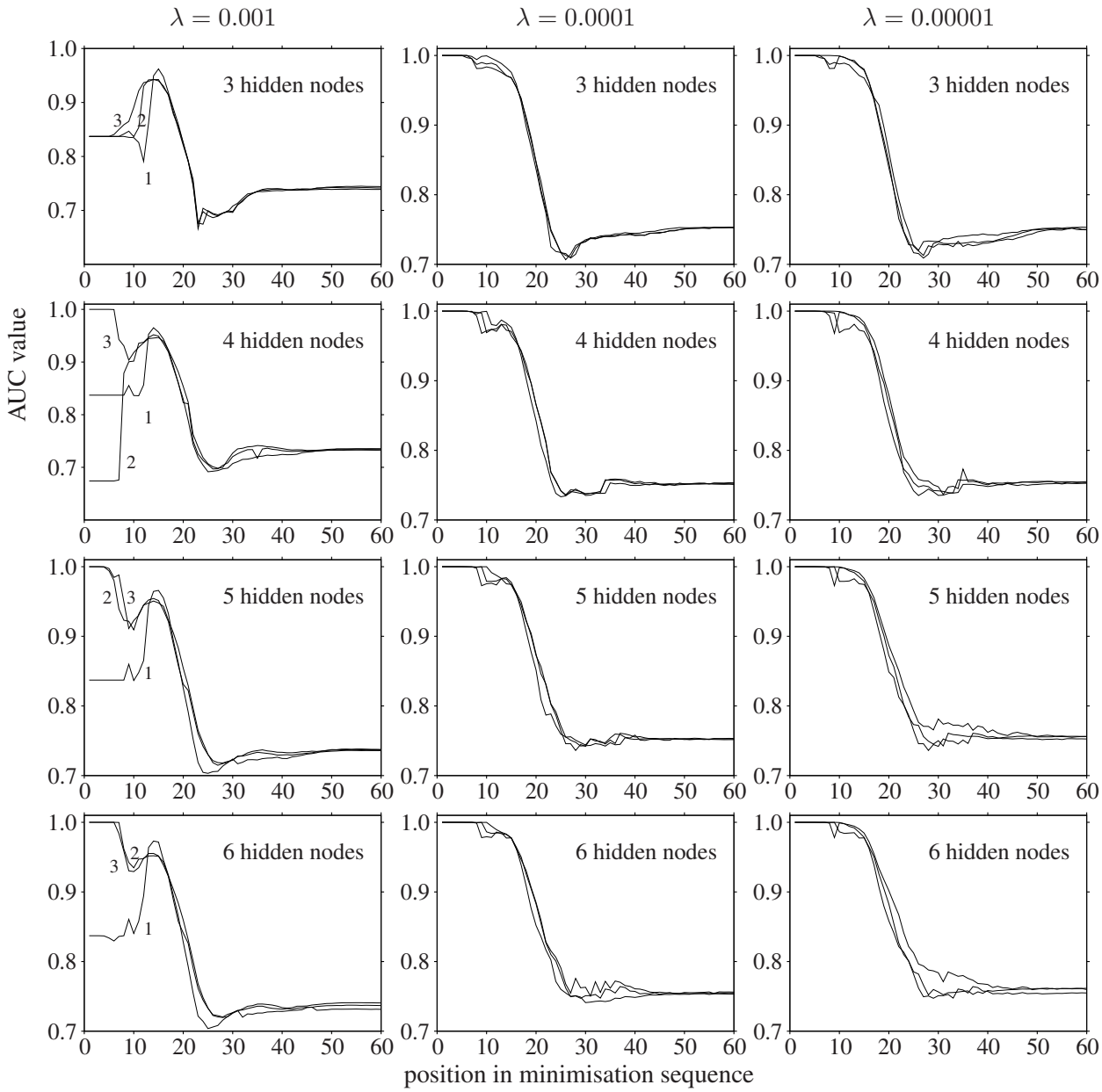

FIG. 1. AUC values for 5,000 minimisation sequences in the LBFGS testing set, in each case evaluated using the parameters obtained for the global minimum neural network fit with 5,000 training sequences. The three columns correspond to regularisation parameters $\lambda = 10^{-3}$, $\lambda = 10^{-4}$, and $\lambda = 10^{-5}$. Results are shown for input data corresponding to the distances r_{12} and r_{13} at one, two and three successive points in each minimisation sequence. The horizontal axis gives the value of the first configuration in the sequence. The curves are labelled with the number of data points used in the first column for $\lambda = 0.001$, where the behaviour close to convergence can be rather different when memory previous configurations is included.

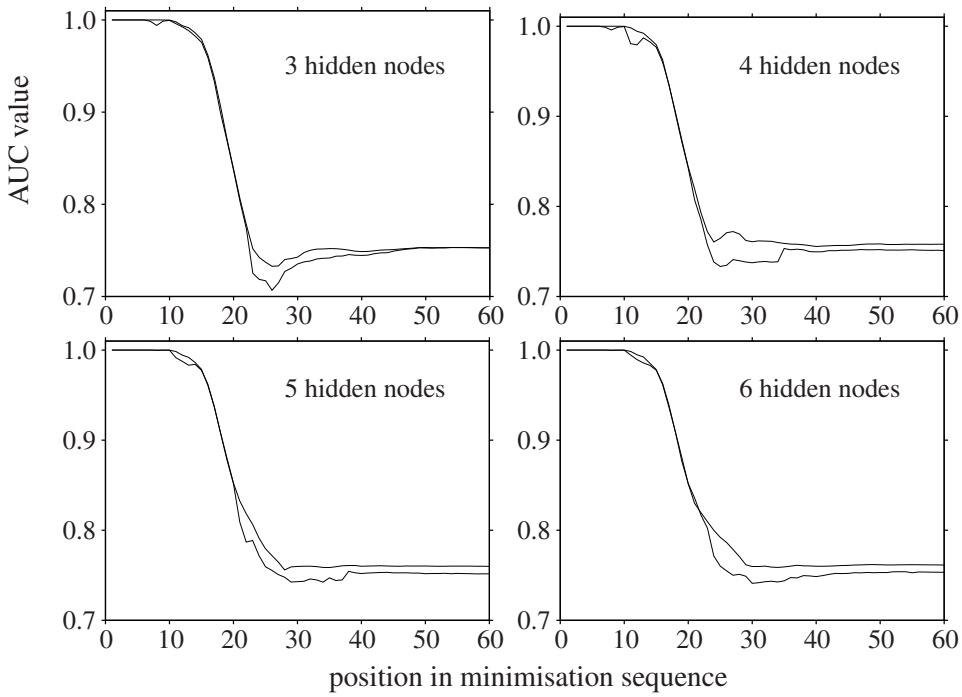

FIG. 2. AUC values for 5,000 minimisation sequences in the LBFGS testing set, evaluated using the parameters obtained for the global minimum neural network fit with 5,000 training sequences and $\lambda = 10^{-4}$. The four panels correspond to 3, 4, 5 or 6 hidden nodes, as marked, with input data for distances r_{12} and r_{13} at a single configuration in each minimisation sequence. Each panel has a second plot of the highest AUC value for the test data attained with any local minimum obtained in training having the same number of inputs and hidden nodes, including results for all the λ values considered and for all values of x from 1 to 80. The AUC value for the global minimum with $\lambda = 10^{-4}$ and the configuration in question is included in this set, but can be exceeded by one of the many local minima obtained over the full range of λ and minimisation sequence data. However, the differences in predictive performance are generally negligible.

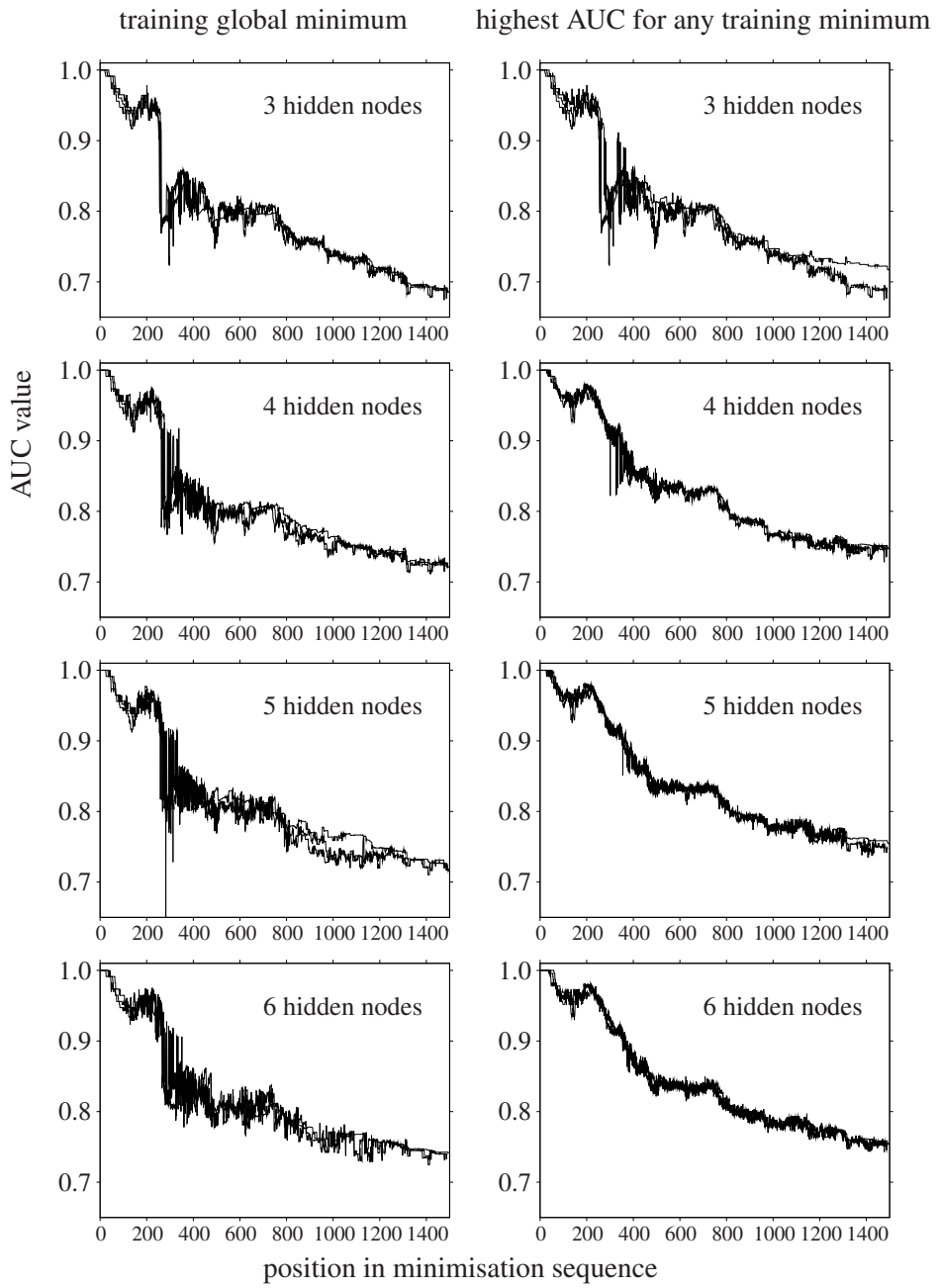

FIG. 3. AUC values for 4,500 minimisation sequences in the testing set corresponding to Bulirsch-Stoer minimisation. The plots in the left column correspond to the parameters obtained for the global minimum neural network fit with 4,500 training sequences. The plots in the right column correspond to the highest AUC value for the testing set for any of the local minima obtained with the training set. Both columns correspond to a regularisation parameter $\lambda = 10^{-5}$. Results are shown for input data corresponding to the distances r_{12} and r_{13} at one, ten and twenty successive points in each minimisation sequence. The horizontal axis gives the value of the first configuration in the sequence. The three plots are similar in each case, and largely superimposed.

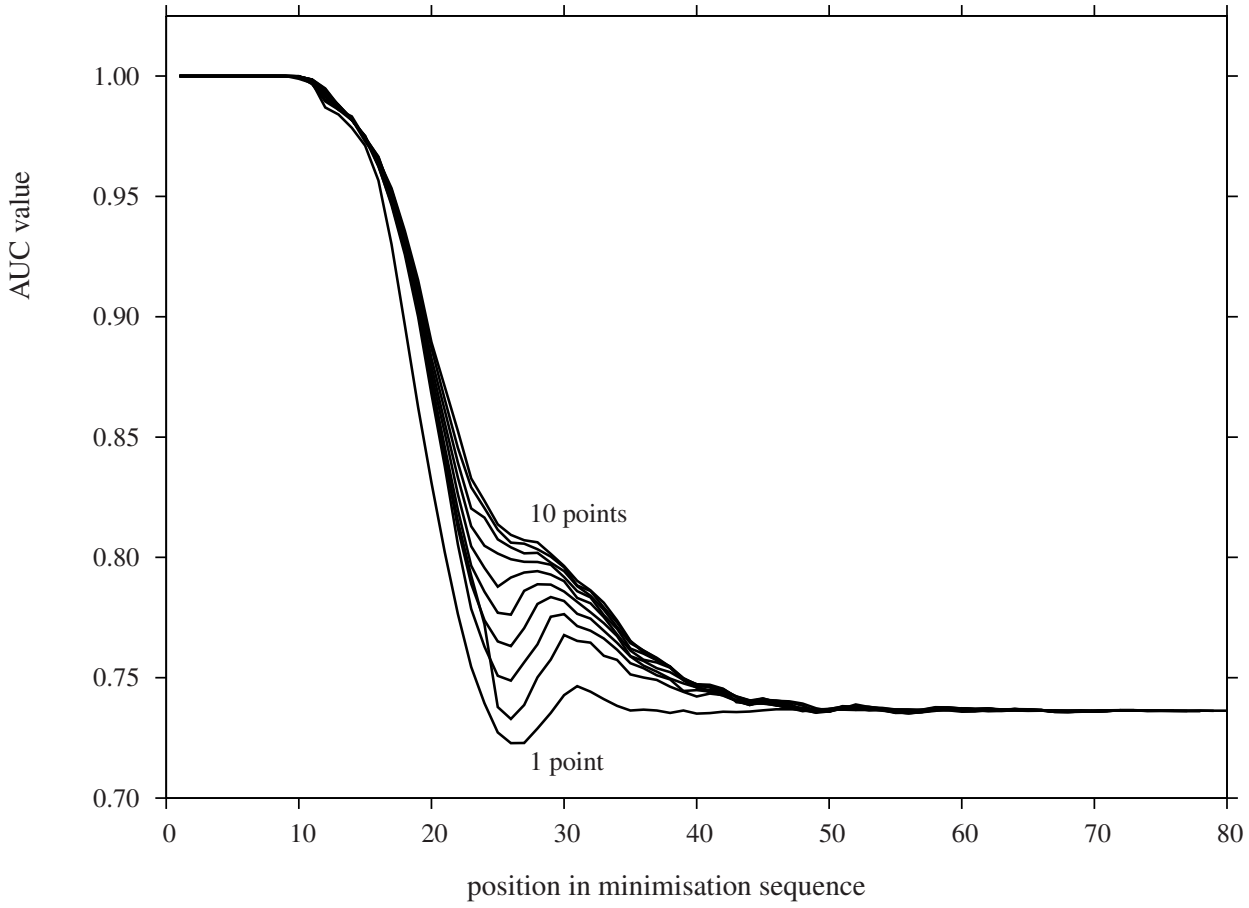

FIG. 4. AUC values for 5,000 minimisation sequences in the LBFGS testing set, evaluated using the parameters obtained for the global minimum quadratic fit with 5,000 training sequences and $\lambda = 10^{-5}$. Results are shown for input data corresponding to the distances r_{12} and r_{13} at between one and ten successive points in each minimisation sequence. The horizontal axis gives the value of the first configuration in the sequence in each case.